

(12) **UK Patent Application** (19) **GB** (11) **2 384 330** (13) **A**

(43) Date of A Publication **23.07.2003**

(21) Application No **0201027.0**

(22) Date of Filing **17.01.2002**

(71) Applicant(s)

**Nick Rayne**  
**187 West Heath Road, LONDON, NW3 7TT,**  
**United Kingdom**

(72) Inventor(s)

**Nick Rayne**

(74) Agent and/or Address for Service

**Lloyd Wise**  
**Commonwealth House,**  
**1-19 New Oxford Street, LONDON,**  
**WC1A 1LW, United Kingdom**

(51) INT CL<sup>7</sup>

**G06F 17/30 , H04Q 7/22**

(52) UK CL (Edition V )

**G4A AUDB**

(56) Documents Cited

**GB 2361840 A**                      **WO 2001/056234 A1**  
**WO 2000/072612 A1**           **US 6321257 B**  
**US 6230004 B**

(58) Field of Search

Other: **ONLINE: EPODOC, WPI, PAJ, INTERNET**

(54) Abstract Title

**System for synchronising a local database using SMS messages**

(57) System allowing a user to interact with user specific data in a database, have means allowing a user to modify specific data in the database and means allowing the user to send a Short Message Service (SMS) message including a predetermined command to a controller associated with the database. The controller includes means for receiving the SMS message, means for determining the command, and means for interacting with the user data according to the command.

**GB 2 384 330 A**

### **Personal Database Access via SMS**

5 In general terms, the invention proposes a method and a system for  
synchronizing a local database with a database server over the Internet;  
querying, deleting, editing, and populating the data via a web browser; and  
querying, deleting, editing, and populating the data via a system of SMS  
messages and responses using a short message server, queuing system, and  
processing application.

10

The invention relates to the fields of telecommunication systems, short  
messaging protocols, short messaging receiving and transmission technology,  
web applications, enterprise messaging and queuing systems, database design,  
database synchronization systems, and local personal computer user interfaces.

15

Although SMS is an extremely popular medium, and there are many applications  
employing it, there is a void in the area of access to user-owned and -managed  
databases presented in a complete solution with an effective querying and editing  
system, with no further development work or integration required.

20

The invention enables simple and complex queries embodied in SMS messages to  
allow access from a mobile telephone to a private or public database from any  
SMS-enabled device. The synchronization component allows the available data to  
be kept as current as is required by the users. The database can be a private  
25 database (such as a Contacts, Schedule or general Personal Information  
Management program), or it can be a corporate database server. The web  
application allows full management of the database from any Internet enabled  
computer, when a user is away from his own computer or his office.

30

An example of a sequence of user actions is as follows:

1. The user downloads a small application, which runs natively on his personal computer from a web site.
2. The user runs the small application using a simple interface, which

synchronizes his database (for example, a contacts and schedule database from a popular Personal Information Management program) with the database held on the server.

3. The user sends an SMS from a mobile terminal to request database information. In the contacts example, the following steps might occur:
- User sends a text message containing "SMITH" to the application telephone number.
  - The SMS server receives this and reads the sender's mobile number.
  - The application queries the database based on the search text and the user mobile number.
  - The application retrieves either a choice of names beginning with "SMITH" or all details stored under that single contact. Further fields can be retrieved using specific commands.
  - The application sends the results back to the user's number via another SMS message.
4. Further commands can be sent along with the search text, which perform a range of functions.
5. Additional management, editing, deleting, and inserting of records can be performed via a standard HTML browser accessing a Servlet which queries the same database.

The dotted line in the diagram below represents the application constituting the invention.

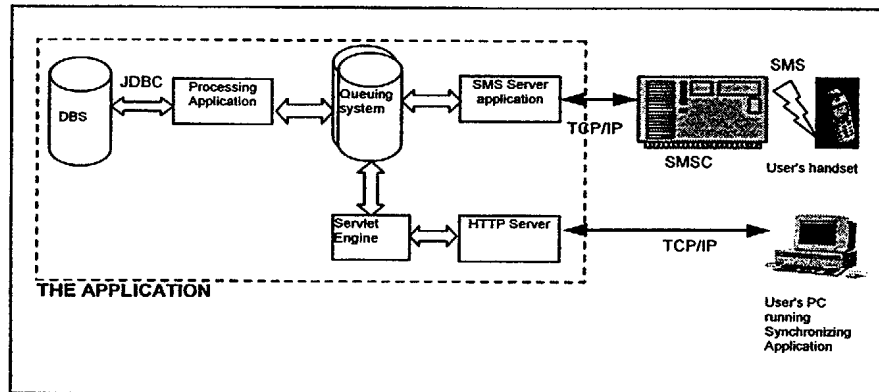


Figure 1 - Overview of Architecture

## 5 **Description of the invention using Contacts example**

- The main application can either reside on a commercial SMSC (Short Messaging Service Centre) or can link to a server via a TCP/IP connection, as illustrated in **Error! Reference source not found. Error! Reference source not found..**
- 10 • The application is associated with a service provider having the SMSC or server and a website for interacting with a user's PC.

The three principal user interactions are detailed below, namely Synchronizing, SMS access, and Browser Access.

### **Synchronization of Database via Internet**

## 15 **User Action**

- User downloads and installs a small application from the service provider web site.
- User clicks on a 'Synchronize' button, either on the Windows task bar, start menu, or desktop, or directly on the Outlook toolbar.

### ***Application Activity***

- The Synchronize application opens an Internet connection to the service provider server for streaming data.
- The Synchronize application exports all records from Outlook, converting to XML format as it does so, and passes the data in a stream to the service provider server.
- At the receiving end, the service provider server reads the XML and imports records into the database. Only those which are more recently changed than the service provider database will overwrite the existing record. If both records have been changed, duplicates will be kept and the user will be informed. These actions are in accordance with standard PIM synchronizing practice.

### **Database Access via SMS**

#### ***User Action***

- 15 The following example indicates how the user interfaces with the TextMine engine: if, for example, one was using the system to store contacts details:

User sends text message      A user sends a simple message to a telephone number associated with the service provider containing search text such as **'JON'**

He receives either:

Result:           **Phil Jones**  
                  **07881111555 mobile**  
                  **02074182000 work**  
                  **02083611234 home**  
                  **Newtown Bank plc**  
                  **336 Blossom Street**  
                  **Newtown NE2 5QT**  
                  **phil@newtown.com**  
                  **SEND M FOR MORE**

or, where there are multiple possible entries:

Alternative       **1 Jonathan Smith**  
result:           **2 Phil Jones**  
                  **3 Suzie Jonstone**  
                  **SEND NUMBER**

in which case the user can send a message containing the digit '1', '2', or '3' to specify the required record.

With most phones, a user can use the results to dial numbers directly without re-entering them using the keypad. For example, Nokia phones have an option to 'Use number' that is, to choose any of the numbers embedded in the text message. This can be saved to the phone memory or dialled.

(Continued...)

## Further Commands

User retrieves a single field      Individual fields from the contacts database can be retrieved by sending, for example, '**N**' (for notes) after the above details are received. The same command can also be sent at the outset as '**JONE N**'. (Sending 'JONE' ensures that only one record is returned). The result would look like this:

Result      **Met Phil at the furniture trade  
fair in Stockholm in Mar 99.  
Reports to John Smith MD.  
Responsible for UK, Ireland,  
France, Italy, and Holland.  
SEND M FOR MORE**

User forwards a contact to another      A contact can be 'bounced' to an acquaintance's phone by sending '**B 07882111555**' or, at the outset, '**JONE B 07882111555**'.  
The acquaintance would receive:

Result            **Phil Jones**  
                      **07881111555 mobile**  
                      **02074182000 work**  
                      **02083611234 home**  
                      **REPLY WITH BLANK MSG TO**  
                      **START FREE TRIAL FROM**  
                      **TEXTMINE.CO.U**

and would subsequently receive details and further instructions by SMS.

Other commands allow the user to populate databases and amend records via SMS. The Nokia Business Card feature, if available, may also be used to add a record. Most data manipulation, however, will be managed via  
 5 the browser interface mentioned above.

### ***Application Activity***

Having seen the process from the user's perspective, the following represents the actions performed by the application:

10

- The SMSC receives SMS messages from mobile phones and directs them to the recipient mobile number (that of the application).
- The SMS Server Application listens for incoming messages using a commercially available JDK.
- 15 • It converts them into Java Message Service (JMS) messages using bespoke developed algorithms, and passes them to an incoming JMS queue (using the standard JMS API), controlled by the Java Enterprise Edition platform.
- The bespoke developed Processing Application picks them off this queue and parses the messages into Search text, commands, and qualifiers, applying  
 20 specialized application logic to create the necessary database queries. For



example, 'JONE B 07882111555' as shown above is interpreted as:

'Perform a search for JONE and send ('bounce') the results to the number shown, instead of the user's number'.

- 5 • The Processing Application then performs the queries, communicating with the central database.
- The results returned from the database are parsed and interpreted by the Processing Application, and any commands embedded in the original message are performed. For example, a long result is split into messages of 140 characters each, leaving room for a message such as 'Send M for more'. An
- 10 'N' command following the search text retrieves only the fields beginning with that letter, such as the Notes field.
- The Processing Application then packages the results into a JMS message and posts them to an outgoing queue, to be collected by the SMS Server Application and sent to the commercial SMSC.
- 15 • The Java Message Service implementation provides resilience since messages will not be lost, even in a complete application failure. Since the same queues are shared by the web application, and indeed by the synchronizing application, the messaging system controls access to the database and avoids the need for additional locking. In addition, most of the different components
- 20 shown above can reside on independent machines. The JMS implementation therefore enables distributed computing as the demands on the application grow, as well as providing future scalability by substitution for commercial middleware components.

### **Database Access via Web Browser**

#### 25 ***User Action***

- A user logs on to the service provider web site using a username and password.
- A user-friendly interface allows him to view, edit, add, and delete records should he be away from his principal PC running the PIM
- 30 application.

### ***Application Activity***

- The Servlet Engine receives requests from a normal web browser via an HTTP Server.
- 5 • It submits these requests to the same incoming queue as used by the SMS messages above.
- The responses returning from the database arrive by way of separate outgoing queue, and subsequently a Response Pool contained within an application Java Bean (not illustrated).

### 10 ***Elements of the Invention***

#### **SMS Server Application**

This performs two principal functions (using a commercially available Java Development Kit) in addition to the conversion function described below:

- 15 • The application listens for incoming SMS messages, and passes them on for conversion into Java Message Service messages.
- It sends messages as they are popped off the queue and converted.

#### **SMS/JMS Conversion**

- 20 This is performed by means of bespoke code, and occurs within the SMS-server component. The routines also push and pop messages to and from the queue.

#### **Queuing Function**

- 25 This is performed by means of bespoke code, but taking advantage of the Java Message Service API, which provides an environment for queues and message entities. The SMS Server, the Servlet Engine, and the Processing Application all perform this function.

### **Processing Application**

This is specially developed by ourselves and performs several functions:

- 5     • It picks messages off the incoming queue.
- It analyzes the mobile originated messages and breaks them down into search, text, commands, and qualifiers.
- It stores session information for users returning with subsequent SMS messages.
- 10    • It also analyzes messages originating from a web browser in a similar way.
- It applies the application logic to both types of messages and performs queries on the database.
- It controls access to the database by both sources, mobile originated
- 15    and web browser originated.
- It processes the returned queries and packages them into messages.
- It delivers the two types of messages to their respective outgoing queues.

### **20     Servlet Engine**

This takes advantage of the Java Servlet API and a commercially available web server. It receives requests from a web page, packages them into Java Messages Service messages for queuing and also extracts outgoing messages from a queue for outward transmission.

25

For example:

1. User sends a web page HTTP request by clicking on a SUBMIT button.
- 30   2. This request is converted into a JMS message containing database instructions. E.g. 'Add new record with data contained herein'.

3. The processing application receives these instructions and accesses the database accordingly.
  - 5 4. The results returned from the database are converted into JMS messages and passed through the application.
  5. The JMS messages are converted into HTTP responses to create the resulting web page visible to the user.
  - 10
- It makes use of its own outgoing queue and a specially developed Response Pool, in the form of an Enterprise Java Bean, to handle the outgoing responses.

#### **Format of the database**

- 15 The database is designed around the core tables being Users, Contacts, and Events. Users is a list of all users, their mobile numbers and log-in details, and various registration details. Contacts contains records for all users, and Events contains schedule events for the Agenda application.

**CLAIMS**

1. A system for a user to interact with user specific data in a database, the system comprising:

5 means for a user to modify user specific data in said database;

means for said user to send a Short Message Service (SMS) message to a controller associated with said database, said message including a predetermined command;

10 said controller comprising means for receiving said SMS message, means for determining said command, and means for interacting with said user data according to said command.

2. A system according to claim 1 wherein said controller further comprises means for sending an SMS message to said user sending  
15 means, said message including user specific data.

3. A system according to claim 1 or 2 wherein the message further comprises one or a combination of the following: search text; user specific data; qualifiers; further predetermined commands.

20 4. A system according to any preceding claim, wherein said modification means is a terminal such as a personal computer coupled to said controller via the Internet.

5. A system according to claim 4, wherein said controller comprises a Web page to interface with said terminal.

6. A system according to any preceding claim, wherein said sending means is a wireless terminal such as a mobile phone or mobile computing device.
7. A system according to claim 3 when dependent on claim 2;  
5 wherein said controller further comprises: means for converting incoming SMS messages to Java Message Service (JMS) messages; means for parsing each said message to determine said command, and any search text or qualifiers in said message; means for querying the database in response to said command and search  
10 text; means for converting the results of said query into a JMS message; means for converting said message into an SMS message.
8. A system as disclosed in any preceding claim wherein said controller further comprises means for sending an SMS message to  
15 a second user sending means, said message including user specific data.
9. A system according to claim 5 wherein said controller further comprises: means for converting web page requests into Java Message Service (JMS) messages containing instructions for  
20 querying the database; means for parsing these messages; means for querying the database in response to said instructions; means for converting the results into JMS messages; and the means for converting said JMS messages into web page responses.
10. A system according to claim 4 comprising means to  
25 synchronize data stored on said personal computer with that stored on said database.



14



INVESTOR IN PEOPLE

**Application No:** GB 0201027.0  
**Claims searched:** 1-10

**Examiner:** Steven Gross  
**Date of search:** 8 March 2002

## Patents Act 1977 Search Report under Section 17

### Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.T):

Int Cl (Ed.7):

Other: Online: EPODOC, WPI, PAJ, Internet

### Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
X	GB 2361840 A (INTELLPROP) See especially page 1 line 23 to page 2 line 18	1-3,6,8
X	WO 01/56234 A1 (SOFTWIRED) See especially page 6 line 22 to page 9 line 17	1-3,6,8-9
X	WO 00/72612 A1 (SOFTWARE.COM) See especially page 3 line 6 to page 6 line 2	1-6,8,10
X	US 6321257 B1 (KOTOLA) See especially column 2 line 13 to column 4 line 17	1-3,6,8
X	US 6230004 B1 (HALL) See especially column 2 lines 19 to 36	1-3,6,8

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.